Changing the *a priori* Model in *globl*

John M. Gipson
May 7, 1993
NVI/GSFC

---

1. Introduction

The sequential least squares analysis program used by the VLBI group at Goddard is called *globl.* This program allows for the determination of station positions and velocities, source position, EOP series, and other parameters of geodetic and geophysical interest. A solution to the least squares problem consists of three parts:

1. In the forward solution, the Combined Global Matrix (CGM) is formed. Essentially, this is the normal equation for the global parameters.
2. At the end of the forward solution, the normal equation is inverted to obtain the global parameters. Typically these are things like source positions, and station positions and velocities.
3. The backward solution solves for the arc parameters. These include nuisance parameters, such as clock and atmosphere variations, as well as parameters of real geophysical interest, such as EOP and nutation offsets.

The middle step takes a few to several minutes. The first and last steps typically take from a few hours to many hours.

In building the CGM *globl* calculates the observed minus the calculated delays (and/or rates). (The so called *"O-C"* values.) The calculated values depend on the particular *a priori* model chosen. In solving the global normal equations, *globl* obtains the corrections to the *a priori* model.

It turns out the changes induced in the normal equations by a change in the *a priori* model are remarkably simple. In fact, the new normal equations can be determined from the original normal equations, the original model, and the new model. Because of this, you only need to perform the forward solution once. Starting with a standard solution, you can generated all other solutions which differ only by a change in the *a priori* model. Not only does this halve the computational time involved in a solution, but it also simplifies the bookkeeping. In comparing the effect of different a priori models, you can now be assured that they were processed in precisely the same way, using the same arcs.

This report derives and describes the algorithm used in changing the *a priori* model. Section 2 briefly discusses the normal equations, and establishes notation. This section also solves the case of unconstrained normal equations. Section 3 discusses the effects of degeneracy in the normal equations, and what happens when constraints are imposed. Section 4 uses the results of sections 2 and 3 to derive the behavior of the normal equations under a change of *a priori* station position, and/or source position. Section 5 has some concluding remarks.

---

2. Unconstrained Normal Equations

The general least squares problem consists of solving the following matrix equation:

(1) $$Na = b$$

Here N is an m by n matrix, a is the n dimensional vector we are solving for, and b is an m dimensional vector which is constructed from the measured data. N is independent of the both the *a priori* model and the measured data, and depends only on the least squares basis functions. The solution to this equation may be over-determined, under-determined, or unique.

In the case of *globl*, N is a square matrix (m=n). The vector a is the difference between the *a priori* model (or value) for the parameters, and their estimated values. Instead of (1), we consider the series of equations:

$$Na_j = b_j$$

which differ only in the choice of the *a priori*. Given a solution a_0 to the normal equations, and the *a priori* model A_0 , the estimated values for the parameters are:

$$A_{tot} = A_0 + a$$

If the solution to the normal equations is unique, this can be written as:

$$A_{tot} = A_0 + N^{-1} b_0$$

where the subscript on b indicates that it depends on the particular model used. Suppose that we change the a priori model.Since a change in the *a priori* model must leave A_{tot} invariant, it follows that:

$$A_0 + N^{-1} b_0 = A_1 + N^{-1} b_1$$

or

(2) $$b_1 = b_0 + N(A_0 - A_1) = b_0 + N\Delta A_{01}$$

Hence if we change the a priori, all we need to do is change the b and this equation tells how to do it. In this equation I have introduced a notation I will use frequently below. Anytime the Δ symbol is used, it will indicate the difference between two vectors, with the first subscript indicating the first vector, and the second subscript the second vector. Hence:

$$A_0 - A_1 = \Delta A_{01} = -\Delta A_{10}$$

3. Constrained Normal Equations

If the normal equations are underdetermined, their solution must be effected by constraining the system. There are many ways of doing so, but essentially they fall into two classes.

In the first method, you impose the constraint while building the normal equations. For example, in *globl*, if you include the line

```
STATIONS YES WESTFORD
```

in the FLAGS section of the control file, the normal equations will be constructed with Westford held fixed at its *a priori* value. The advantage of this approach is that the normal equations are, by construction, invertible.

In the second method the constraint is imposed after the normal equations are built, and just prior to inverting them. The advantage of this method is that it is more general. If you decide that you want to keep Richmond fixed instead of Westford, or impose some other constraints to remove the translational degeneracy from the VLBI solutions, you don't have to generate a whole new solution. In fact, *globl* can operate in this mode with respect to station position, and most recent solutions are generated in this mode.

A further advantage of the second mode is that the effect of a change in the *a priori* model in the normal equations is given by Eq. (2), where the new *b* vector is constructed prior to imposing the constraints required to invert the normal equation.

Since the older solutions were not generated in this mode, and in any case, *globl* does not operate in this mode for all parameters, we need to know how to deal with the first method of imposing constraints. My approach will be to start with the unconstrained equation, and then impose the constraints to see what happens.

Suppose that the normal equations are degenerate. Mathematically, this means that the normal matrix has zero eigenvalues. Suppose that it has *m* such *zero* eigenvalues. Let us denote the vectors associated with these eigenvalues by Z_α .Then, by definition

$$NZ_\alpha = 0$$

The choice of the *Z's* is not important as long as they span the null space of *N*. (That is, an eigenvector that is annhihlated by *N* can be written as a linear combination of the *Z's*.) The existence of the zero eigenvectors means that the solution to the normal equations is not unique. If *a* is one solution to the normal equations, then so is:

$$a + \sum_{\alpha=1}^{m} \lambda_\alpha Z_\alpha$$

Note that the α label the vector, and not the components. (All of the *Z's* are n dimensional vectors.) Physically, the *Z's* are related to the symmetries of the system. In the case of VLBI, these are the transformations of the global parameters that leave the delay and rates invariant.

Instead of a single solution to the normal equations, we have a whole family of solutions parameterized by the λ's. In what follows, assume that *a* is some fixed (but as yet unspecified) solution to the normal equations. To uniquely fix the solution, we must impose *m* constraints which will determine the λ's. Without loss of generality, assume that the constraints we will impose will be to make the adjustments to the first *m* parameters of the *a priori* model vanish. We can always rearrange the parameters, and perform a similarity (generalized coordinate transform) to make this so.

Effectively, we have split the parameter space into two parts: The constrained portion, which we will refer to as the "constraint space", denoted with a subscript *c*, and the restricted space, denoted with a subscript *r*. The normal equations can then be written as:

$$\begin{pmatrix} N_c & N_{cr} \\ N_{cr}^T & N_r \end{pmatrix} \begin{pmatrix} a_c + \lambda Z_c \\ a_r + \lambda Z_r \end{pmatrix} = \begin{pmatrix} b_c \\ b_r \end{pmatrix}$$

For simplicity the summation on the *Z's* is implied. The reduced normal equations are:

$$N_r(a_r + \lambda Z_r) = b_r$$

or

(3) $$N_r a_r = b_r - N_r(\lambda Z_r) = b_r(\lambda)$$

(Recall that $a_c + \lambda Z_c$ vanishes because of the constraint.) The $N\lambda Z$ term on the right hand side of this equation arises from the constraints. In the last equality I have defined a new quantity $b_r(\lambda)$ which is essentially the *"O-C"* in the presence of the constraints.

A change in the *a priori* model will induce a change in the normal equations. It is useful to partition this effect into two pieces: The change in the *a priori* confined to the "constraint space", and the change in the model in the "restricted space." From the considerations of the previous section, we can immediately write down the change in the *b* due to a change in the *a priori* model in the restricted subspace:

(4) $$b_{1,r} = b_{0,r} + N_r(A_{0,r} - A_{1,r}) = b_{0,r} + N_r \Delta A_{01,r}$$

The effect of a change in the *a priori* model in the constrained subspace is somewhat more complicated. To start with, note that the two solutions must be related by a different choice of λ's. In fact, we must have:

$$\lambda_1 Z_c = \lambda_0 Z_c + (A_{1,c} - A_{0,c})$$

or:

$$\Delta\lambda_{01} Z_c = \Delta A_{01,c}$$

Since the *Z's* span the null space, this equation can be solved for the difference in λ's. Reintroducing the summation operator which was dropped above, we have:

(5) $$\sum_{\alpha=1}^{m} Z_{\alpha j,c} \Delta\lambda_{01,\alpha} = \Delta A_{01,j,c}$$

Recall that the index α labels the vectors, while *j* labels the coordinates of the vector's. Now, $Z_{\alpha j,c}$ can be viewed as an *m* by *m* matrix. Doing so, we can formally solve for the δλ:

(6) $$\Delta\lambda_{01} = Z_c^{-1} \Delta A_{01,c}$$

and hence

(7) $$b_{1,r} = b_{0,r} + N_r \Delta\lambda_{01} Z_r$$

Equations 6 and 7 gives the general solution to the change in the restricted normal equations due to a change in the constrained portion of the *a priori* model. The somewhat complicated form of these equations has a simple physical interpretation in all the specific cases we have examined.

---

4. Some Examples of Constrained Systems.

In the previous section we derived the general equations for how the change of the *a priori* model affects the normal equation. In this section we will look at a few concrete examples, where we will limit our discussion to a change in the *a priori* model in the constrained space. The simpler consequences of a change in the *a priori* model in the restricted space leads to Eq. (4) above. In the general situation in which the *a priori* model changes in both spaces, you need to add the two changes.

Station Position Constrained.

It is well known that the VLBI observables are independent of the absolute position of the stations. A simultaneous translations of all stations will leave the delay and the rate invariant. This leads to a degeneracy in the normal equations. One way of removing this degeneracy is to fix the position of one of the stations. For simplicity, assume that we are only solving for *n* station position, that we fix the first station, and that the station parameters are in the natural order:

$$X \equiv (x_1, y_1, z_1, x_2, y_2, z_2, \ldots x_n, y_n, z_n)$$

The zero eigenvectors of the normal equations are given by:

$$Z_x = (1,0,0,1,0,0,\ldots 1,0,0) \quad Z_y = (0,1,0,0,1,0,\ldots 0,1,0) \quad Z_z = (0,0,1,0,0,1,\ldots 0,0,1)$$

Note that under a simultaneous translation of all station positions we have:

$$X \to X + \delta x Z_x + \delta y Z_y + \delta z Z_z$$

The constraint space corresponds to the first three coordinates. The Z matrix is a 3 by 3 identity matrix. It follows from this that the change in the normal equations induced by a change in the *a priori* model is given by:

$$b_{1r} = b_{0r} + N_r \delta A_{10,r} + N_r(\overrightarrow{Z}_r \bullet \overrightarrow{\Delta x_c})$$

where δx_c is the change in the *a priori* position of the first (constrained) station.

Right Ascension Constrained

A more complicated example arises if we solve for both source and station positions, and fix the right ascension of the source. The reason we need to fix the RA is that the VLBI observables are invariant with respect to rotation. In particular, if we translate the RA of all sources by a fixed amount, and rotate the station positions about the z axis by the same amount, the VLBI delay and rate will remain unchanged.

For simplicity, assume that we constrain the right ascension of a single source, but leave the station positions unconstrained.(In the next section we will see what happens if we fix a source *and* a station.) Furthermore, assume that the order of the parameters is *m* sources, followed by *n* stations:

$$A_0 = (RA_1, DEC_1, RA_2, DEC_2, \ldots RA_m, DEC_m, x_1, y_1, z_1, \ldots, x_n, y_n, z_n)$$

The zero eigenvector of the normal matrix corresponding to rotation around the z axis is given by:

$$Z_\omega = (1, 0, 1, 0, \ldots 1, 0, y_1, -x_1, 0, \ldots, y_n, -x_n, 0)$$

Under an infinitesimal rotation $\delta\omega$ the source and station coordinates transform according to:

$$A_0 \to A_0 + \Delta\omega Z_\omega$$

The constrained subspace consists of just the right ascension of the first source. The residual space is the remaining source and station parameters. It follows that a change δRA in right ascension of the first source will induce a change in the *b* vector of amount:

$$b_{1r} = b_{0r} + \Delta RA\ N_r Z_{\omega,r} = b_{0r} + \Delta RA\ N_r (0, 1, 0, \ldots 1, 0, y_1, -x_1, 0, \ldots, y_n, -x_n, 0)$$

Fixing Station Position and Right Ascension

Our final example will involve fixing both the station position and the right ascension. For the purposes of this example, assume that the order of the coordinates are the first station and first source, followed by the remaining sources, followed by the remaining stations:

$$(x_{1,}y_{1,}z_{1,}RA_1, DEC_1, RA_2, DEC_2, ...RA_{m,}DEC_{m,}x_2, y_{2,}z_2, x_3, y_{3,}z_3..., x_n, y_{n,}z_n)$$

There are four zero eigenvectors associated with this: the three translations, and a rotation:

$$\begin{aligned}
Z_x = &\ (1, 0, 0, \quad 0, 0, \quad 0, 0, \quad ... \quad 0, 0, \quad 1, 0, 0, \quad 1, 0, 0, \quad ... \quad 1, 0, 0) \\
Z_y = &\ (0, 1, 0, \quad 0, 0, \quad 0, 0, \quad ... \quad 0, 0, \quad 0, 1, 0, \quad 0, 1, 0, \quad ... \quad 0, 1, 0) \\
Z_z = &\ (0, 0, 1, \quad 0, 0, \quad 0, 0, \quad ... \quad 0, 0, \quad 0, 0, 1, \quad 0, 0, 1, \quad ... \quad 0, 0, 1) \\
Z_\omega = &\ (-y_{1,}x_{1,}0 \quad 1, 0, \quad 1, 0, \quad ... \quad 1, 0, \quad -y_{2,}x_{2,}0, \quad -y_{3,}x_{3,}0, \quad ... \quad -y_{n,}x_{n,}0)
\end{aligned}$$

The first four parameters are the constrained to be 0. It follows that:

$$\begin{aligned}
Z_{x,c} = &\ (1, 0, 0, 0) \\
Z_{y,c} = &\ (0, 1, 0, 0) \\
Z_{z,c} = &\ (0, 0, 1, 0) \\
Z_{\omega,c} = &\ (-y_{1,}x_{1,}0, 1)
\end{aligned}$$

Equation 6 becomes:

$$(\Delta\lambda_x - y_1\Delta\lambda_\omega, \Delta\lambda_y + x_1\Delta\lambda_y, \Delta\lambda_z, \Delta\lambda_\omega) = (\Delta x_{1,}\Delta y_{1,}\Delta z_1, \Delta RA_1)$$

This can be solved for the δλ's to yield:

$$\begin{aligned}
\Delta\lambda_x = &\ \Delta x_1 - y_1\Delta RA_1 \\
\Delta\lambda_y = &\ \Delta y_1 + x_1\Delta RA_1 \\
\Delta\lambda_z = &\ \Delta z_1 \\
\Delta\lambda_\omega = &\ \Delta RA_1
\end{aligned}$$

The physical effect of performing the transformation given by:

$$\Delta\lambda \bullet \vec{Z}$$

is to rotate all of the sources positions by δRA_1 and to rotate all of the stations by a same amount about the first station. This is in contrast to the previous example, where all stations were rotated about the origin.

5. Concluding Remarks: General Constraints and Symmetries

In the previous section, we looked at three concrete examples of constrained normal equations. In all three cases, we needed to apply constraints because of an underlying symmetry in the system. By symmetry, I mean some transformation that leaves the VLBI observables invariant. These symmetries, the constraints, and the effect of the constraints all had simple physical interpretations. In concluding, I want to momentarily return to the more general situation.

In the general case, suppose that the observables are invariant with respect to an *m* parameter family of transformations of the *a priori* model specified by (λ_α, α=1,m). Explicitly:

$$A = A(\lambda)$$

Under an infinitesimal transformation $\delta\lambda$, we have:

$$A_j(\delta\lambda) = A_j(0) + \sum_{\alpha=1}^{m} \delta\lambda_\alpha \frac{\partial A_j}{\partial \lambda_\alpha} = A_j(0) + \overrightarrow{\delta\lambda} \bullet \frac{\partial A_j}{\partial \overrightarrow{\lambda}}$$

The index *j* labels the parameter, and the α the transformation. Since the observables remain invariant under this transformation, it follows that the normal equations do as well. In particular, the zero eigenvectors of the normal equations are given by:

$$Z_\alpha = \frac{\partial A}{\partial \lambda_\alpha}$$

where I have suppressed the coordinate index. This equation tells us how to obtain the zero eigenvectors of the normal equation by studying the symmetries of the observables. All of the eigenvectors of the previous section can be seen as obtained as special cases of this general formula.

When we constrain the system, we fix the $\delta\lambda$. To determine the effect of the constraint on the normal equations when we change the *a priori* model, we first determine what is the $\delta\lambda$ that will carry one *a priori* model into another. Once this is determined, the effect on the normal equations is given by:

$$\delta b = N_r \overrightarrow{\delta\lambda} \bullet \frac{\partial A_j}{\partial \overrightarrow{\lambda}}$$

Here N_r is the projection of *N* on the restricted subspace.

In analyzing the effect of a constraint, the hard work comes in determining what symmetry the constraint breaks, i.e., what is the invariance that exists without the constraint. Once this is done the effect on the normal equations can be immediately written down.